

1. A nmea 0183 sentence transporter over ethernet for a nmea 0183 sentence transporter over ethernet for increasing transmission speed, allowing for dynamic routing and complete transparency to the user. comprising:

means for converting the electrical signals from rs422 to microprocessor level signal levels and then from the microprocessor signal levels to rs422 electrical signal standards;

means for processing the software instructions for transporting the nmea 0183 sentences and the routing instructions, electrically connected to said means for converting the electrical signals from rs422 to microprocessor level signal levels and then from the microprocessor signal levels to rs422 electrical signal standards;

means for providing the compatablity with ethernet signals levels and the microprocessor signal levels, electrically connected to said means for processing the software instructions for transporting the nmea 0183 sentences and the routing

instructions;

means for indentifying input processing of the nmea 0183 data;

means for polling routine that is to cycle through each of the input ports looking for nmea 0183 sentences, logically linked to said means for indentifying input processing of the nmea 0183 data;

means for a checking routine that filters out nmea 0183 sentences that are not to be transported to the ethernet, logically linked to said means for polling routine that is to cycle through each of the input ports looking for nmea 0183 sentences;

means for adding a prefix header to the ethernet sentence packet, logically linked to said means for a checking routine that filters out nmea 0183 sentences that are not to be transported to the ethernet;

means for building an ethernet data packet comprised of the header fields and the nmea 0183 sentence, logically linked to said means for adding a prefix header to the ethernet sentence packet;

means for processing the ethernet data packet by submitting it to the output driver, logically linked to said means for building an ethernet data packet comprised of the header fields and the nmea 0183 sentence;

means for writing the nmea 0183 data sentence to the correct output port;

means for routing the nmea 0183 data packet to the correct rs422 port, logically linked to said means for writing the nmea 0183 data sentence to the correct output port;

means for checking data filters to see if nmea 0183 sentence data should be sent to output ports, logically linked to said means for routing the nmea 0183 data packet to the correct rs422 port;

means for stripping off headers from the ethernet nmea 0183 data sentence packet., logically linked to said means for checking data filters to see if nmea 0183 sentence data should be sent to output ports;

means for parsing the packet to determine the

routing for the nmea 0183 sentence, logically linked to said means for stripping off headers from the ethernet nmea 0183 data sentence packet.;

means for polling the ethernet packet looking for nmea 0183 encapsulated data, logically linked to said means for parsing the packet to determine the routing for the nmea 0183 sentence;

means for defining data which makes up the udp header, including the ip address;

means for defining data which encapsulates the prefix of the nmea 0183 sentence data, containing but not limited to indentifications, and routing;

means for defining data which makes up the standard nmea 0183 sentence;

means for defining the data suffix added to the nmea 0183 sentence;

means for defining the data field that makes up the unique indetifier for the ethernet data packet;

means for defining the data field which defines the origin id fof the data sentence;

means for defining the data field of the destination id for the data packet;

means for defining the data field of the destination ip address for the ethernet data packet;

means for defining the data field of the destination port address for the ethernet data packet;

means for defining the data field that which is made up of the nmea 0183 sentence code for this ethernet data packet;

means for reading the ethernet packet of data from the transparency program running on the computer;

means for checking headers for routing and mapping instructions, logically linked to said means for reading the ethernet packet of data from the transparency program running on the computer;

means for stripping off the header from the ethernet packet, logically linked to said means for checking headers for routing and mapping instructions;

means for routing and writting of the nmea 0183 sentence to the correct virtual com port, logically linked to said means for stripping off the header from the ethernet packet;

means for reading from the virtual com port and looking for a valid nmea 0183 sentence;

means for checking the routing and mapping table to determine what header intructions to build, logically linked to said means for reading from the virtual com port and looking for a valid nmea 0183 sentence;

means for constructing the header on the front of the nmea 0183 sentence which contains the routing and indentification information, logically linked to said means for checking the routing and mapping table to determine what header intructions to build;

means for writting the nmea 0183 packet containing headers and trailers to the ethernet, logically linked to said means for constructing the header on the front of the nmea 0183 sentence which contains the routing and indentification information;

means for defining the data that is used for ethernet transmission of the nmea 0183 data sentence; and

means for interfacing between the software interface utility and the user applications.

2. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for converting the electrical signals from rs422 to microprocessor level signal levels and then from the microprocessor signal levels to rs422 electrical signal standards comprises a RS422 electro static protection rs422_transceiver.

3. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for processing the software instructions for transporting the nmea 0183 sentences and the routing instructions

comprises a Microprocessor capable of interfacing to RS422 transceiver and Ethernet transceiver electrically, Microprocessor capable of being programmed with logic as stated microprocessor.

4. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for providing the compatablity with ethernet signals levels and the microprocessor signal levels comprises an Electrically compatible interface between microprocessor and standard Ethernet signal levels ethernet_transceiver.

5. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for indentifying input processing of the nmea 0183 data comprises a nmea_0183_rs422_inputs.

6. The nmea 0183 sentence transporter over ethernet in

accordance with claim 1, wherein said means for polling routine that is to cycle through each of the input ports looking for nmea 0183 sentences comprises a poll_rs422_inputs.

7. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for a checking routine that filters out nmea 0183 sentences that are not to be transported to the ethernet comprises a filter_packets.

8. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for adding a prefix header to the ethernet sentence packet comprises an add_prefix.

9. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for building

an ethernet data packet comprised of the header fields and the nmea 0183 sentence comprises a build_output_packet.

10. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for processing the ethernet data packet by submitting it to the output driver comprises an output_packet.

11. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for writing the nmea 0183 data sentence to the correct output port comprises a nmea_0183_rs422_outputs.

12. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for routing the nmea 0183 data packet to the correct rs422 port comprises a route_to_port.

13. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for checking data filters to see if nmea 0183 sentence data should be sent to output ports comprises a check_filters.

14. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for stripping off headers from the ethernet nmea 0183 data sentence packet. comprises a strip_headers.

15. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for parsing the packet to determine the routing for the nmea 0183 sentence comprises a parse_packet.

16. The nmea 0183 sentence transporter over ethernet in

accordance with claim 1, wherein said means for polling the ethernet packet looking for nmea 0183 encapsulated data comprises a receive_ethernet_packet.

17. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining data which makes up the udp header, including the ip address comprises an udp_header.

18. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining data which encapsulates the prefix of the nmea 0183 sentence data, containing but not limited to indentifications, and routing comprises an ethernet_header.

19. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining

the data suffix added to the nmea 0183 sentence comprises an ethernet_trailer.

20. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining the data field that makes up the unique identifier for the ethernet data packet comprises an ethernet_identifier.

21. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining the data field which defines the origin id for the data sentence comprises an origin_id.

22. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining the data field of the destination id for the data packet comprises a destination_id.

23. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining the data field of the destination ip address for the ethernet data packet comprises an ip_address.

24. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining the data field of the destination port address for the ethernet data packet comprises a port_address.

25. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining the data field that which is made up of the nmea 0183 sentence code for this ethernet data packet comprises a sentence_codes.

26. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for reading the ethernet packet of data from the transparency program running on the computer comprises a receive_packet.

27. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for checking headers for routing and mapping instructions comprises a check_header.

28. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for stripping off the header from the ethernet packet comprises a strip_headers_from_packet.

29. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for routing and writting of the nmea 0183 sentence to the correct

virtual com port comprises a route_to_virtual_com_port.

30. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for reading from the virtual com port and looking for a valid nmea 0183 sentence comprises a read_from_virtual_com_port.

31. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for checking the routing and mapping table to determine what header instructions to build comprises a check_routing.

32. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for constructing the header on the front of the nmea 0183 sentence which contains the routing and identification information comprises a build_headers.

33. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for writting the nmea 0183 packet containing headers and trailers to the ethernet comprises a write_packet.

34. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for defining the data that is used for ethernet transmission of the nmea 0183 data sentence comprises an ethernet_nmea_0183_data_packet.

35. The nmea 0183 sentence transporter over ethernet in accordance with claim 1, wherein said means for interfacing between the software interface utility and the user applications comprises a virtual_com_ports.

36. A nmea 0183 sentence transporter over ethernet for a

nmea 0183 sentence transporter over ethernet for increasing transmission speed, allowing for dynamic routing and complete transparency to the user. comprising:

a RS422 electro static protection rs422_ transceiver, for converting the electrical signals from rs422 to microprocessor level signal levels and then from the microprocessor signal levels to rs422 electrical signal standards;

a Microprocessor capable of interfacing to RS422 transceiver and Ethernet transceiver electrically, Microprocessor capable of being programmed with logic as stated microprocessor, for processing the software instructions for transporting the nmea 0183 sentences and the routing instructions, electrically connected to said RS422_ Transceiver;

an Electrically compatible interface between microprocessor and standard Ethernet signal levels ethernet_transceiver, for providing the compatibility with ethernet signals levels and the

microprocessor signal levels, electrically connected to said Microprocessor;

a nmea_0183_rs422_inputs, for indentifying input processing of the nmea 0183 data;

a poll_rs422_inputs, for polling routine that is to cycle through each of the input ports looking for nmea 0183 sentences, logically linked to said NMEA_0183_RS422_Inputs;

a filter_packets, for a checking routine that filters out nmea 0183 sentences that are not to be transported to the ethernet, logically linked to said Poll_RS422_Inputs;

an add_prefix, for adding a prefix header to the ethernet sentence packet, logically linked to said Filter_Packets;

a build_output_packet, for building an ethernet data packet comprised of the header fields and the nmea 0183 sentence, logically linked to said Add_Prefix;

an output_packet, for processing the ethernet data packet by submitting it to the output driver,

logically linked to said Build_Output_Packet;

 a nmea_0183_rs422_outputs, for writing the
nmea 0183 data sentence to the correct output port;

 a route_to_port, for routing the nmea 0183
data packet to the correct rs422 port, logically
linked to said NMEA_0183_RS422_Outputs;

 a check_filters, for checking data filters to
see if nmea 0183 sentence data should be sent to
output ports, logically linked to said Route_to_Port;

 a strip_headers, for stripping off headers
from the ethernet nmea 0183 data sentence packet.,
logically linked to said Check_Filters;

 a parse_packet, for parsing the packet to
determine the routing for the nmea 0183 sentence,
logically linked to said Strip_Headers;

 a receive_ethernet_packet, for polling the
ethernet packet looking for nmea 0183 encapsulated
data, logically linked to said Parse_Packet;

 an udp_header, for defining data which makes
up the udp header, including the ip address;

an ethernet_header, for defining data which encapsulates the prefix of the nmea 0183 sentence data, containing but not limited to indentifications, and routing;

a nmea_0183_sentence, for defining data which makes up the standard nmea 0183 sentence;

an ethernet_trailer, for defining the data suffix added to the nmea 0183 sentence;

an ethernet_identifier, for defining the data field that makes up the unique indetifier for the ethernet data packet;

an origin_id, for defining the data field which defines the origin id fof the data sentence;

a destination_id, for defining the data field of the destination id for the data packet;

an ip_address, for defining the data field of the destination ip address for the ethernet data packet;

a port_address, for defining the data field of the destination port address for the ethernet data packet;

a sentence_codes, for defining the data field that which is made up of the nmea 0183 sentence code for this ethernet data packet;

a receive_packet, for reading the ethernet packet of data from the transparency program running on the computer;

a check_header, for checking headers for routing and mapping instructions, logically linked to said Receive_Packet;

a strip_headers_from_packet, for stripping off the header from the ethernet packet, logically linked to said Check_Header;

a route_to_virtual_com_port, for routing and writting of the nmea 0183 sentence to the correct virtual com port, logically linked to said Strip-Headers_From_Packet;

a read_from_virtual_com_port, for reading from the virtual com port and looking for a valid nmea 0183 sentence;

a check_routing, for checking the routing and mapping table to determine what header intructions

to build, logically linked to said Read_From_Virtual_COM_Port;

a build_headers, for constructing the header on the front of the nmea 0183 sentence which contains the routing and identification information, logically linked to said Check_Routing;

a write_packet, for writting the nmea 0183 packet containing headers and trailers to the ethernet, logically linked to said Build-Headers;

an ethernet_nmea_0183_data_packet, for defining the data that is used for ethernet transmission of the nmea 0183 data sentence; and

a virtual_com_ports, for interfacing between the software interface utility and the user applications.

37. The nmea 0183 sentence transporter over ethernet as recited in claim 36, further comprising:

an user_applications, for applications that are user supplied needing access to the nmea 0183

data and operating on the computer, logically connected to said Route_to_Virtual_COM_Port, and logically connected to said Read_From_Virtual_COM_Port.

38. A nmea 0183 sentence transporter over ethernet for a nmea 0183 sentence transporter over ethernet for increasing transmission speed, allowing for dynamic routing and complete transparency to the user. comprising:

a RS422 electro static protection rs422_ transceiver, for converting the electrical signals from rs422 to microprocessor level signal levels and then from the microprocessor signal levels to rs422 electrical signal standards;

a Microprocessor capable of interfacing to RS422 transceiver and Ethernet transceiver electrically, Microprocessor capable of being programmed with logic as stated microprocessor, for processing the software instructions for

transporting the nmea 0183 sentences and the routing instructions, electrically connected to said RS422_Transceiver;

an Electrically compatible interface between microprocessor and standard Ethernet signal levels ethernet_transceiver, for providing the compatibility with ethernet signals levels and the microprocessor signal levels, electrically connected to said Microprocessor;

a nmea_0183_rs422_inputs, for indentifying input processing of the nmea 0183 data;

a poll_rs422_inputs, for polling routine that is to cycle through each of the input ports looking for nmea 0183 sentences, logically linked to said NMEA_0183_RS422_Inputs;

a filter_packets, for a checking routine that filters out nmea 0183 sentences that are not to be transported to the ethernet, logically linked to said Poll_RS422_Inputs;

an add_prefix, for adding a prefix header to the ethernet sentence packet, logically linked to

said Filter_Packets;

a build_output_packet, for building an ethernet data packet comprised of the header fields and the nmea 0183 sentence, logically linked to said Add_Prefix;

an output_packet, for processing the ethernet data packet by submitting it to the output driver, logically linked to said Build_Output_Packet;

a nmea_0183_rs422_outputs, for writing the nmea 0183 data sentence to the correct output port;

a route_to_port, for routing the nmea 0183 data packet to the correct rs422 port, logically linked to said NMEA_0183_RS422_Outputs;

a check_filters, for checking data filters to see if nmea 0183 sentence data should be sent to output ports, logically linked to said Route_to_Port;

a strip_headers, for stripping off headers from the ethernet nmea 0183 data sentence packet., logically linked to said Check_Filters;

a parse_packet, for parsing the packet to

determine the routing for the nmea 0183 sentence,
logically linked to said Strip-Headers;

 a receive_ethernet_packet, for polling the
ethernet packet looking for nmea 0183 encapsulated
data, logically linked to said Parse_Packet;

 an udp_header, for defining data which makes
up the udp header, including the ip address;

 an ethernet_header, for defining data which
encapsulates the prefix of the nmea 0183 sentence
data, containing but not limited to indentifications,
and routing;

 a nmea_0183_sentence, for defining data which
makes up the standard nmea 0183 sentence;

 an ethernet_trailer, for defining the data
suffix added to the nmea 0183 sentence;

 an ethernet_identifier, for defining the data
field that makes up the unique indetifier for the
ethernet data packet;

 an origin_id, for defining the data field
which defines the origin id fof the data sentence;

 a destination_id, for defining the data field

of the destination id for the data packet;

an ip_address, for defining the data field of the destination ip address for the ethernet data packet;

a port_address, for defining the data field of the destination port address for the ethernet data packet;

a sentence_codes, for defining the data field that which is made up of the nmea 0183 sentence code for this ethernet data packet;

a receive_packet, for reading the ethernet packet of data from the transparency program running on the computer;

a check_header, for checking headers for routing and mapping instructions, logically linked to said Receive_Packet;

a strip_headers_from_packet, for stripping off the header from the ethernet packet, logically linked to said Check_Header;

a route_to_virtual_com_port, for routing and writting of the nmea 0183 sentence to the correct

virtual com port, logically linked to said Strip-Headers_From_Packet;

an user_applications, for applications that are user supplied needing access to the nmea 0183 data and operating on the computer, logically connected to said Route_to_Virtual_COM_Port;

a read_from_virtual_com_port, for reading from the virtual com port and looking for a valid nmea 0183 sentence, logically connected to said User_Applications;

a check_routing, for checking the routing and mapping table to determine what header instructions to build, logically linked to said Read_From_Virtual_COM_Port;

a build_headers, for constructing the header on the front of the nmea 0183 sentence which contains the routing and identification information, logically linked to said Check_Routing;

a write_packet, for writting the nmea 0183 packet containing headers and trailers to the ethernet, logically linked to said Build-Headers;

an ethernet_nmea_0183_data_packet, for
defining the data that is used for ethernet
transmission of the nmea 0183 data sentence; and
a virtual_com_ports, for interfacing between
the software interface utility and the user
applications.